

Rád bych zde poděkoval vedoucímu mé bakalářské práce Ing. Petru Navrátilovi za odborné vedení, cenné připomínky a rady, jež mi v průběhu práce poskytoval.

Prohlašuji, že jsem na celé diplomové práci pracoval samostatně a použitou literaturu jsem citoval.

Ve Zlíně, 31. 05. 2005

.....

jméno diplomanta

Abstrakt

Podstatou této bakalářské práce je vypracování elektronického manuálu pro System Identification Toolbox v prostředí HTML stránek. Cílem je tedy vypracovat návod, aby případný zájemce neznalý této problematiky byl schopen alespoň částečně pracovat s tímto toolboxem bez toho, aniž by musel studovat různé referenční manuály a mohl si vybrat tu metodu, která pro jeho daný problém poskytne ty nejlepší výsledky.

Abstract

The aim of this final work is to develop an electronic manual for the System Identification Toolbox in the environment of HTML pages. The objective therefore is to create an instructions manual for a possible applicant not familiar with this problematics so that he is able to at least partially use this toolbox without having to study various reference manuals, and can choose a method allowing the best results for his given problem.

OBSAH

ÚVOD.....	5
1 TEORETICKÁ ČÁST.....	6
1.1 CHARAKTERISTIKA IDENTIFIKACE	6
1.2 TYPY MATEMATICKÝCH MODELŮ	7
1.3 ANALYTICKÁ IDENTIFIKACE	9
1.4 EXPERIMENTÁLNÍ IDENTIFIKACE.....	10
1.4.1 Klasifikace podle druhu modelu.....	12
1.4.2 Klasifikace podle způsobu zpracování výsledku experimentu	14
1.4.3 Klasifikace podle druhu testovacího signálu	15
1.4.4 Klasifikace podle kritéria kvality identifikace.....	17
2 UŽIVATELSKÝ NÁVOD.....	20
2.1 POPIS SYSTEM IDENTIFICATION TOOLBOXU	21
2.2 OPERACE S DATY	22
2.2.1 LOAD	22
2.2.2 SAVE.....	23
2.2.3 DETREND	23
2.3 PARAMETRICKÝ ODHAD	23
2.3.1 AR	24
2.3.2 ARX.....	24
2.3.3 ARMAX	30
2.3.4 IVAR	33
2.3.5 IV4.....	34
2.3.6 OE.....	35
2.3.7 BJ.....	36
2.3.8 PEM.....	37
2.4 NEPARAMETRICKÝ ODHAD	38
2.4.1 CRA.....	39
2.4.2 ETFE	41
2.4.3 SPA.....	41
2.5 ODHAD STRUKTURY MODELU	43
2.5.1 STRUC	43
2.5.2 ARXSTRUC	43
2.5.3 IVSTRUC	44
2.5.4 SELSTRUC	44
2.5.5 Sett.....	44
2.6 KONTROLA MODELU	45
2.6.1 IDSIM.....	45
2.6.2 Compare	45
2.7 ZOBRAZENÍ MODELU.....	46
2.7.1 BODEPLOT	46
2.7.2 FFLOT.....	46
2.7.3 IDPLOT.....	46
2.7.4 PRESENT.....	47
2.7.5 ZPLOT	47
2.8 KONVERZE MODELU.....	48
2.8.1 C2D	48
2.8.2 D2C	48
2.8.3 TH2FF	48
2.8.4 ZPKDATA	49
ZÁVĚR.....	50

SEZNAM POUŽITÝCH ZDROJŮ	51
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	52
SEZNAM OBRÁZKŮ.....	53

ÚVOD

V překotně se vyvíjejícím světě výpočetní techniky a programového vybavení mají od prvopočátku nezastupitelné místo programy pro numerické výpočty. Z počátku byla podpora uživatele realizována matematickými knihovnami pro obecné programovací jazyky. Později se vyvinuly samostatné programy, které je možné rozdělit asi do tří oblastí – tabulkové procesory s rozšířenými matematickými funkcemi, specializované programy na řešení určitého okruhu problémů a profesionální univerzální matematické programy, do kterých patří Matlab.

Pro řešení úkolů v oboru automatizace existuje vynikající programové prostředí firmy Mathworks nazvané MATLAB. Toto prostředí má modulový charakter, to znamená, že lze dokoupit moduly tzv. toolboxy pro jednotlivé oblasti. MATLABu a jednotlivé toolboxy se neustále vyvíjejí. Firma Mathworks nabízí komplexní servis a na svých [www stránkách](#) má také diskusní forum, kde lze jednotlivé problémy řešit s odborníky takřka z celého světa.

Pro usnadnění řešení problémů z oblasti experimentální identifikace je možné použít System Identification Toolbox.

Podstatou této bakalářské práce je vypracování elektronického manuálu pro System Identification Toolbox v prostředí HTML stránek. Cílem je tedy vypracovat návod, aby případný zájemce neznalý této problematiky byl schopen alespoň částečně pracovat s tímto toolboxem bez toho, aniž by musel studovat různé referenční manuály a mohl si vybrat tu metodu, která pro jeho daný problém poskytne ty nejlepší výsledky.

1 TEORETICKÁ ČÁST

1.1 Charakteristika identifikace

Proces identifikace se definuje jako určení systému na základě vstupů a výstupů a to tak, aby určený systém byl na základě vhodně zvolených experimentů shodný se systémem zkoumaným. Určený systém budeme nazývat modelem. Matematický model je vhodnou formou pro vyjádření poznatků o zkoumaných objektech. Při sestavování modelu se reálný objekt zjednodušuje tím, že se zanedbají vlastnosti nepodstatného významu, které by dělaly model složitějším, ale na druhou stranu musí být do modelu zahrnuty všechny charakteristické vlastnosti zkoumaného procesu.

Cílem identifikace je tedy vytvořit takový model, aby jeho chování bylo v jistém smyslu stejné, jako chování reálného systému. Při procesu identifikace hraje významnou roli informace o objektu, protože jestli ta není správná nebo není úplná, tak ani výsledný model nemůže být správný. Informace o objektu se nazývá aposteriorní a byla získána na základě jeho pozorování. Poznatky nahromaděné při pozorování celé skupiny objektů se nazývají apriorní. Tyto poznatky jsou pak uspořádány v ucelený soubor a představují bohatší informaci. Úspěšnost procesu modelování a identifikace závisí na několika faktorech, které jsou: vhodný výběr apriorní informace, aposteriorní informace, vlastní identifikační algoritmus a nakonec srovnání modelu se skutečností.

Proces identifikace můžeme chápat jako ztotožnění našich poznatků o zkoumaném objektu s reálným systémem, protože se jedná o poznávací proces záleží na pozorovateli a cílu identifikace. Výsledkem tohoto procesu je poznání, které je závislé na schopnostech pozorovatele a je formulované pomocí matematických vztahů.

Na poznávací proces je možno pohlížet jako na získávání informací a následné působení pozorovatele na identifikovaný objekt.

Identifikace systémů nalézá široké uplatnění v mnoha oblastech. V řízení a systémovém inženýrství jsou metody identifikace použity pro získávání vhodných modelů pro syntézu regulátoru, návrh prediktivního řízení nebo simulace. V aplikacích zabývajících se zpracováním signálu jsou modely získané pomocí identifikace systému použity pro spektrální analýzu, odhalení poruch, rozpoznávání obrazců, adaptivní filtraci. Metody

identifikace systémů jsou též úspěšně využívány i v netechnických oborech jako jsou biologie, ekologie a ekonomika.

1.2 Typy matematických modelů

Podle charakteru dělíme matematické modely na dvě velké skupiny:

- *dynamické* - vazbu mezi vstupními a výstupní veličinami představují diferenciální nebo diferenční rovnice.
- *statické* - závislost mezi vstupem a výstupem je popsána algebraickými rovnicemi.

Podle toho, zda jsou parametry dynamických modelů závislé na čase rozlišujeme modely:

- *nestacionární* – časové závislé
- *stacionární* - časové nezávislé

Z hlediska linearity můžeme provést dělení modelů na:

- *lineární*
- *nelineární*

Z hlediska charakteru vazby mezi vstupem a výstupem se modely dělí na:

- *vnější*
- *vnitřní*

Jako vnějších modelů se používá diferenciálních a diferenčních rovnic, impulsních nebo přechodových charakteristik.

Podle struktury a účelu, ke kterému má model sloužit se setkáváme s modely:

- *parametrickými*
- *neparametrickými*

Neparametrické modely vyjadřují relaci mezi závisle a nezávisle proměnou, nevyžadují žádné informace o struktuře identifikovaného systému.

Jako představitel neparametrických modelů je model ve tvaru integrálu:

$$y(t) = \int_0^{\infty} g(\tau)u(t-\tau)d\tau \quad (1)$$

$g(\tau)$ - impulsní funkce

Výhodou parametrických modelů je jejich snadné modelování, protože pomocí malého objemu údajů o parametrech identifikovaného systému snadno popíšeme jeho dynamické chování. Zápornou stránkou je nutnost znalosti struktury systému.

Typickým představitelem parametrických modelů je stacionární lineární diferenciální rovnice:

$$a_n y^n(t) + \dots + a_1 y'(t) + a_0 y(t) = b_m u^m(t) + \dots + b_1 u'(t) + b_0 u(t) \quad (2)$$

$$m \leq n$$

Nebo diferenční tvar:

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = b_1 u(k-1) + \dots + b_m u(k-m) \quad (3)$$

Popřípadě lineární stavový model:

$$\begin{aligned} x(t) &= Ax'(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \quad (4)$$

Při identifikaci se budeme snažit získat modely v parametrickém tvaru a v případě neparametrických modelů provedeme jejich parametrizaci.

Z hlediska způsobu identifikace rozdělujeme modely na:

- *analytické*
- *experimentální*

S ohledem na chování procesu, který probíhá ve vyšetřovaném objektu, můžeme experimentální modely rozdělit na:

- *deterministické*
- *stochastické*

Deterministický model získáme tak, že na vstup vyšetřovaného objektu přivedeme přesně definované testovací signály a na objekt nebudou působit žádné poruchy. Naproti tomu většina reálných objektů má stochastický tvar. To znamená, že výstup soustavy není dán jen vstupním signálem, ale je ovlivněn i náhodnými veličinami. Tyto náhodné děje

mohou probíhat jak uvnitř objektu, tak mohou přicházet z vnějšího okolí. Při matematickém popisu takového jevu nevystačíme s klasickými deterministickými modely, ale je nutné použít obecnější pravděpodobnostní přístup.

Posledním větším dělením matematických modelů je dělení na modely typu:

- *online*
- *offline*

Jednorázové (offline) modely jsou vhodné pro stacionární objekty, kdežto průběžné (online) modely jsou na základě neustále prováděných experimentů zpřesňovány, a proto jsou vhodné pro objekty s nestacionárním chováním.

1.3 Analytická identifikace

Při analytickém způsobu identifikace sestavujeme matematický model na základě matematicko-fyzikální analýzy objektu.

Prvním krokem při matematicko-fyzikální analýze je výběr souboru vstupních a výstupních veličin, pomocí kterých je možno s určitou přesností popsat daný proces. Důležité je vybrat podstatné veličiny. Je třeba se vyvarovat zanedbání podstatných veličin, což by vedlo k neadekvátnosti modelu reálného objektu.

Druhým krokem matematicko-fyzikální analýzy je rozdělení technologického procesu na základní články a sestavení systému dynamických rovnic popisujících jejich dílčí vlastnosti. Z takového základního modelu odvozujeme modely v různých formách podle požadavků syntézy řízení. Zákony, které nejčastěji používáme při analýze, jsou:

- zákon typu zachování
- zákon typu sdílení
- stavové rovnice
- bilance entropie

Posledním krokem je tzv. specifikace modelu procesu, tj. určení hodnot neznámých parametrů a koeficientů odvozeného systému závislostí (rovníc), které určují platnost modelu pro daný objekt v konkrétních podmínkách.

Matematický model systému získaný teoretickou analýzou je třeba vždy upravit do tvaru, z kterého by byly zřejmé dynamické vlastnosti vyšetřovaného objektu. Je nutné model zjednodušit – aproximovat. Původní matematické vztahy se nahrazují jednoduššími a získává se model v použitelnějším tvaru. Při této aproximaci však hrozí, že výsledný model nebude adekvátní objektu.

Výhodné je řešit diferenciální rovnice na osobním počítači. Musíme však provést vhodnou matematickou úpravu, ve které převedeme diferenciální rovnice n -tého řádu na systém n rovnic prvního řádu.

1.4 Experimentální identifikace

Hlavní charakteristikou experimentální identifikace je, že využívá údaje a informace o zkoumaném objektu v průběhu jeho pozorování. Rozborem průběhu vstupních a výstupních veličin identifikovaného objektu získáme matematický model, který vyjadřuje vnější popis systému. Tento model nám však nic neříká o vnitřní struktuře identifikovaného objektu.

Při experimentální identifikaci vycházíme z dostupných apriorních informací a ze znalosti zkoumaného objektu. Veličiny, se kterými při identifikaci pracujeme rozdělujeme na:

- vstupní veličiny u
- výstupní veličiny y
- externí poruchy v
- náhodné poruchy n

První etapou experimentální identifikace je získání co nejobsáhlejších informací o vyšetřovaném objektu. Tyto informace mohou pocházet buď z praktických zkušeností nebo z matematicko-fyzikální analýzy. Podle získaných informací volíme druh a parametry vstupního signálu, periodu vzorkování, délku měření, druhy a rozsahy měřících přístrojů atd.

Další fází experimentu je vlastní měření prováděné na identifikovaném objektu. Změřené průběhy před vlastním vyhodnocením upravíme. Pokud použijeme počítač nejen ke zpracování, ale také ke snímání dat, identifikační proces se výrazně zrychlí.

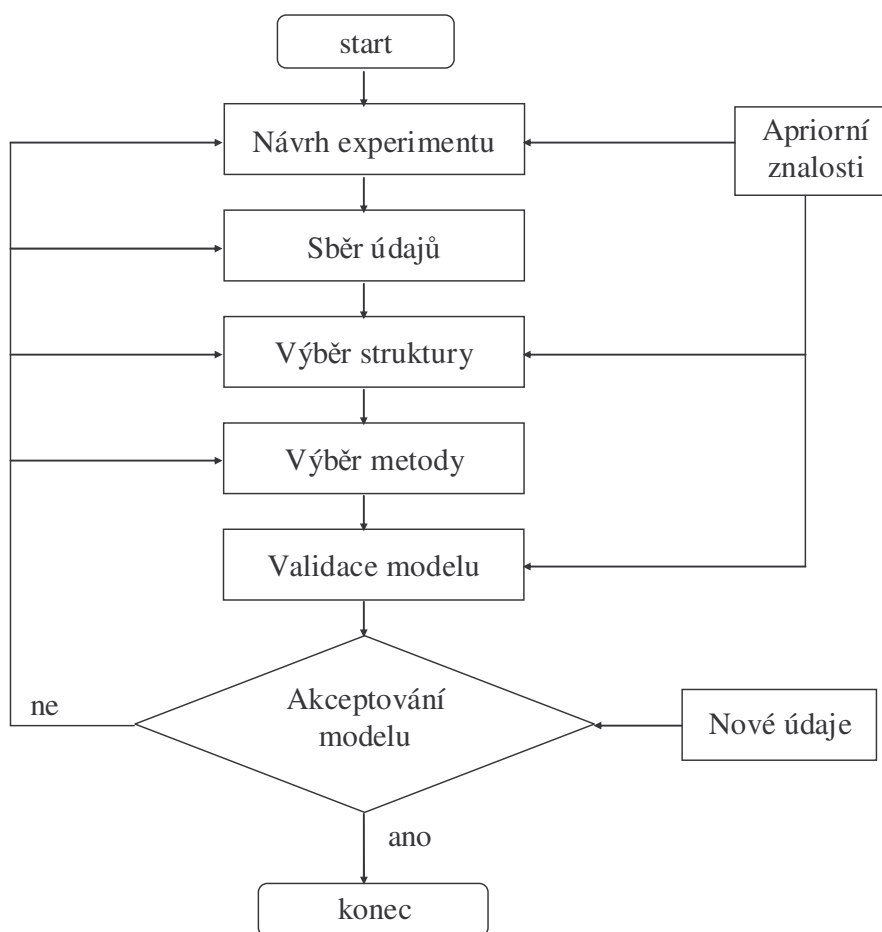
Naměřená data slouží buď pro odhad parametrů modelu s předem známou strukturou nebo se struktura modelu odhaduje společně s parametry. Pokud struktura modelu není známa, tak se vypočítají parametry několika variant modelů s rozdílnou strukturou a testováním se vybere model s nejnižším počtem parametrů, u kterého složitější struktura nepřináší výrazné zlepšení přesnosti. Je výhodné vyhodnotit data několika identifikačními metodami a metody vzájemně porovnat.

Model sestavený pomocí experimentální identifikace je nutno verifikovat, to znamená kontrolovat s počátečními znalostmi a předpoklady, analyzovat jeho přesnost a posoudit, zda je vhodný pro daný účel. Pokud model neodpovídá reálnému objektu, je nutno experiment opakovat nebo provést nové vyhodnocení. Proces kontroly modelu je jedna z nejdůležitějších a nejnáročnějších operací celé experimentální identifikace.

Do experimentální identifikace, na rozdíl od analytické, lze zahrnout spoustu závažných faktorů. Je to dáno hlavně tím, že na většinu reálných objektů působí náhodné veličiny, nebo měřené veličiny jsou ovlivněné náhodnými chybami.

K nevýhodám experimentální identifikace patří to, že identifikovaný objekt musí být přístupný experimentu a sledované veličiny musí být měřitelné. Nezanedbatelným faktem je ta skutečnost, že z modelu získaného experimentální identifikací nemůžeme získat informace o vnitřní struktuře zkoumaného objektu.

Během vývoje teorie identifikace a modelování byla navržena řada metod experimentální identifikace. Z počáteční snahy o co nejjednodušší postup při experimentální identifikaci se s rozvojem výpočetní techniky přechází na stále složitější metodiku výpočtů parametrů modelu. Různorodost identifikačních metod spočívá ve volbě vstupního testovacího signálu, ve volbě typu matematického modelu a jeho struktury, ve volbě kritéria pro shodu modelu a identifikovaného objektu a podle způsobu zpracování výsledků experimentu. Podle těchto kritérií je provedeno rozdělení metod v následujících kapitolách.



Obrázek 1. Postup při experimentální identifikaci

1.4.1 Klasifikace podle druhu modelu

Jedním z hlavních měřítek při klasifikaci identifikačních metod je volba druhu modelu, neboť ke zvolenému modelu většinou přísluší i určitý druh signálu a určité kritérium kvality identifikace nebo identifikačního algoritmu.

Použitý model by měl splňovat požadavky:

- umožnit jednoduché řešení při vyhodnocování naměřených hodnot
- co nej přesněji popsat dynamické vlastnosti identifikovaného objektu
- měl by v sobě zahrnovat vliv rušivých signálů ,které působí na zkoumaný objekt

Obecně můžeme modely rozdělit na modely se strukturou (parametrické) a modely bez struktury (neparametrické). Pod pojmem struktura modelu označujeme tvar modelu, zejména pokud se týká jeho řádu a velikosti dopravního zpoždění.

Parametrický model stejného druhu mění svoji strukturu pro různé druhy identifikovaných objektů. U metod pracujících s parametrickými modely je nutné strukturu modelu buď předem znát nebo ji současně s parametry modelu odhadnout.

U neparametrických metod jsou výsledky identifikačního experimentu ve formě tabulky nebo grafu. Neparametrické modely nemůžeme přímo použít pro řízení, protože většina řídicích strategií je založena na znalosti parametrů modelu. Proto musíme pro účely řízení neparametrické modely vhodným způsobem parametrizovat.

Nejčastěji používané typy modelů si nyní podrobně popíšeme:

a) Model ve tvaru diferenciální rovnice nebo operátorového přenosu

Diferenciální rovnice má tvar:

$$a_n y^n(t) + \dots + a_1 y'(t) + a_0 y(t) = b_m u^m(t) + \dots + b_1 u'(t) + b_0 u(t) \quad (5)$$

$$m \leq n$$

Modely ve tvaru diferenciální rovnice popisují závislost mezi spojitými vstupními a výstupními signály objektu. Protože většina identifikovaných objektů je spojitá, odpovídá často tento tvar modelu analytickým modelům, takže koeficienty modelu představují fyzikální parametry objektu. Do této skupiny patří především jednoduché metody identifikace, jakými jsou vyhodnocování impulsních charakteristik, aproximace přechodových charakteristik, metoda postupné integrace apod., které jsou vhodné pro identifikaci deterministických objektů.

b) Model ve tvaru diferenční rovnice nebo diskrétního přenosu

Diferenční rovnice má tvar:

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = b_1 u(k-1) + \dots + b_m u(k-m) \quad (6)$$

Diferenční rovnice popisuje závislost mezi diskrétními hodnotami vstupních a výstupních veličin objektu. Převodem spojitých veličin na diskrétní přechází matematický popis z diferenciální rovnice na rovnici diferenční. Derivace jsou v diferenční rovnici nahrazeny diskrétními hodnotami časové funkce v čase, což přináší značné zjednodušení numerického řešení odhadu parametrů.

Model ve tvaru diferenční rovnice se obvykle používá u statických a pravděpodobnostních metod odhadu parametrů, kdy výpočet je natolik složitý, že se neobejde bez výpočetní techniky. Je-li výsledků z experimentální identifikace použito

k verifikaci modelu získaného analytickým způsobem, převádí se diferenční rovnice na rovnici diferenciální.

c) Model ve tvaru frekvenční charakteristiky

Frekvenční charakteristika popisuje chování objektu při různých frekvencích a sice pomocí absolutní hodnoty poměru výstupního a vstupního harmonického signálu a pomocí jejich fázového posunu. Klasický způsob proměřování objektů poruchovým signálem s proměnlivou frekvencí je pro některý druh fyzikálních veličin běžný, pro jiné fyzikální veličiny je však technicky nerealizovatelný.

Frekvenční charakteristiku je také možno získat výpočtem s použitím výkonové spektrální hustoty. Nevýhodou modelu ve tvaru frekvenční charakteristiky je, že se obtížně převádí na jiný typ modelu, vhodnější pro další využití.

d) Model ve tvaru impulsní charakteristiky

Impulsní charakteristiku lze buď získat přesným měřením odezvy na vstupní signál ve tvaru impulsu nebo vyhodnocením vstupního a výstupního signálu identifikovaného objektu korelační analýzou.

Přímé zavádění impulsu na vstup je možné jen u objektů bez šumových signálů, neboť odezva má relativně malou amplitudu. Výhodnější je metoda korelační analýzy, kterou je možno použít i pro objekty, na které působí náhodný šum. Vstupní signál se v tom případě volí jako náhodný nebo pseudonáhodný.

1.4.2 Klasifikace podle způsobu zpracování výsledku experimentu

Výsledkem experimentálního měření jsou buď spojité záznamy vstupních signálů a jim odpovídajících odezev nebo diskrétní hodnoty těchto signálů měřené v ekvidistantních časových intervalech. Metody experimentální identifikace můžeme rozdělit podle toho, zda se při nich zpracovávají spojité nebo diskrétní signály.

U jednodušších metod, mezi které patří například analýza přechodových charakteristik, se používá vyhodnocení spojitých signálů. Pro vyhodnocování naměřených dat na číslicovém počítači je naopak nutné zadávat data v diskrétním tvaru, takže v případě spojitého záznamu se tento vzorkuje s určitou periodou.

Dalším hlediskem pro rozdělení metod podle způsobu zpracování naměřených hodnot je dělení na metody explicitní a implicitní. Explicitní metody zpracovávají všechna data

najednou až po ukončení měření. Nevýhodou explicitních metod je, že všechny výsledky měření je nutno až do vyhodnocení uchovávat. Počet naměřených hodnot je omezen možnostmi použité techniky.

Implicitní metody zpracovávají data postupně tak, jak byly v časovém sledu naměřeny. Z několika málo diskrétních hodnot se vytvoří model, který se dalšími změřenými hodnotami neustále zpřesňuje. Při tomto způsobu identifikace není nutno všechna naměřená data uchovávat.

1.4.3 Klasifikace podle druhu testovacího signálu

Vstupní testovací signály můžeme třídit z různých hledisek. V prvním případě rozlišujeme přirozené provozní signály pozorované v průběhu provozu a uměle vytvořené vstupní signály s určitými vlastnostmi, přiváděné na vstup. Vlastnosti signálů posuzujeme podle jejich charakteristik časových, frekvenčních, statistických a jiných. Identifikovaný objekt je buzen vstupním signálem a z odezvy se snažíme analyzovat vlastnosti zkoumaného objektu. Vzniknutí do podstaty vlastností je v experimentu značně závislé na způsobu buzení identifikovaného objektu a výběr typu testovacího signálu. Signály tedy můžeme třídit na:

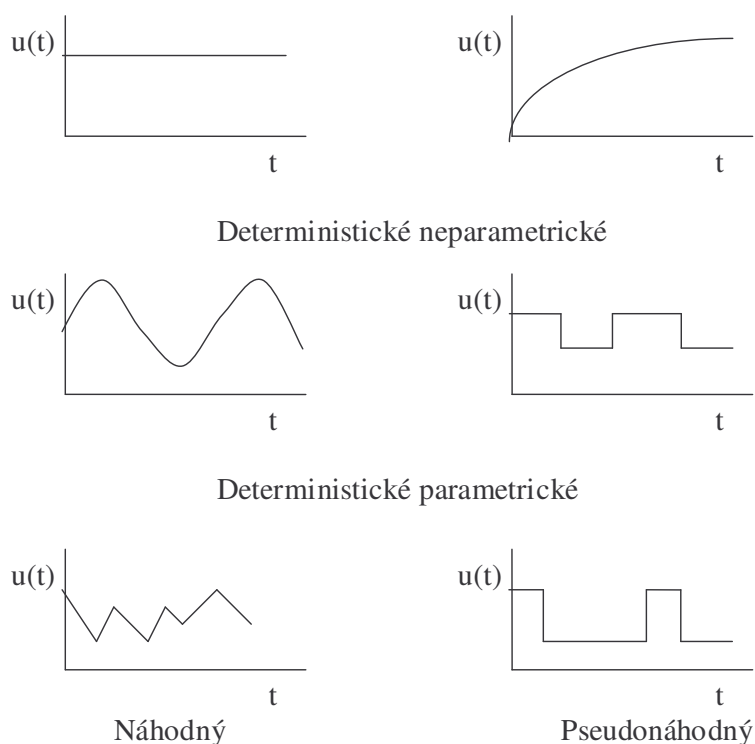
- *deterministické signály* – jejich průběhy v čase známe, lze tedy určit jejich hodnoty pro každý časový okamžik.
- *náhodné signály* – jejich průběhy v čase jsou náhodnými funkcemi času, můžeme určit jen jejich statistické charakteristiky.
- *pseudonáhodné signály* – průběhy v čase jsou známy, v rámci jedné periody mají charakter známé realizace náhodného procesu, tyto realizace se periodicky opakují.

V těchto třídách jsou možné další klasifikace:

Deterministické signály – neperiodické a periodické

Náhodné signály – stacionární a nestacionární

Pseudonáhodné signály – jednohladinové a vícehladinové



Obrázek 2. Příklady různých typů vstupních signálů

Aperiodickým deterministickým signálem je například jednotkový impuls, pravoúhlý impuls, lichoběžníkový impuls, skokový funkce a periodickým deterministickým signálem je například harmonický průběh, pravoúhlý průběh, lichoběžníkový průběh.

Při výběru testovacích signálů musíme vzít v úvahu hledisko praktické realizovatelnosti. Celá řada testovacích signálů má pouze teoretický charakter a prakticky je můžeme realizovat jen s určitým stupněm přiblížení. Typickými představiteli takových signálů jsou jednotkový impuls, skoková funkce či bílý šum.

Deterministické vstupní testovací signály charakterizujeme především jejich spektrální vlastnostmi a to spojitým spektrem u aperiodických a diskretním spektrem u periodických signálů.

Kvalitu testovacího vstupního signálu posuzujeme podle toho, v jakém rozsahu zkoumá vlastnosti identifikovaného objektu a v jaké šířce spektra umožňuje prověřit přenosové vlastnosti objektů. Z toho vyplývají požadavky na volbu testovacího signálu. Vhodným signálem bude ten, jehož spektrální vlastnosti překrývají celý rozsah frekvenčního spektra identifikovaného objektu. Při volbě testovacího signálu pro identifikaci reálného objektu

jsme omezení možností výběru jen z určitých tříd signálů, které jsou pro objekt z bezpečnostních a technologických důvodů přípustné a technicky realizovatelné.

Mezi nejčastější metody využívající aperiodický deterministický vstupní signál patří vyhodnocování přechodových a impulsních charakteristik. Výhoda těchto metod spočívá v relativně krátké době měření odezev a v jejich poměrně jednoduchém vyhodnocování. Nevýhodou aperiodických deterministických vstupních signálů je ta skutečnost, že mají ve spektru obsaženy v menší míře vysoké frekvence, takže u objektů s rušivou složkou výstupního signálu může být výrazně zhoršena přesnost modelu.

Periodické deterministické signály se používají při proměřování frekvenčních charakteristik identifikovaných objektů. Vyžadují poměrně dlouhou dobu měření, protože tato měření se musí opakovat pro různé frekvence signálu. Výhodou identifikace pomocí deterministických signálů je reprodukovatelnost experimentu.

Metody sloužící k vyhodnocování odezev na náhodné a pseudonáhodné signály jsou vhodné zejména pro identifikaci stochastických objektů, ale lze je samozřejmě použít i u soustav deterministických. Tyto metody jsou založeny na statickém vyhodnocování naměřených hodnot, a proto vyžadují poměrně dlouhou dobu měření. To platí zejména pro vyhodnocování odezev na náhodné signály. Vyhodnocování naměřených hodnot se u těchto metod obvykle neobejde bez výpočetní techniky.

1.4.4 Klasifikace podle kritéria kvality identifikace

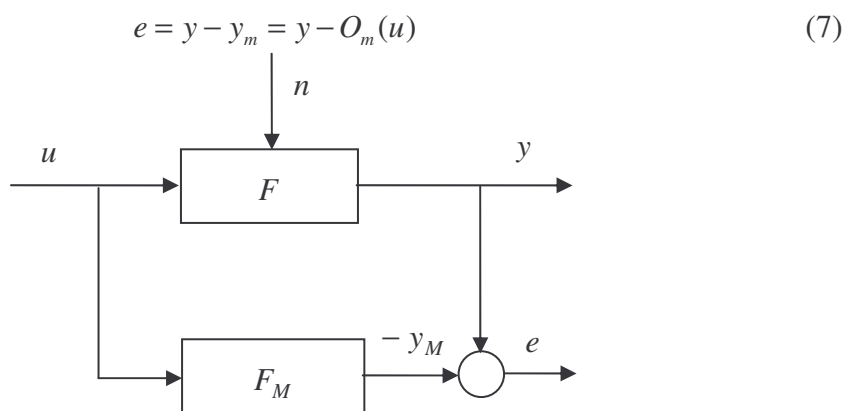
Při známé struktuře modelu shodu modelu (operátoru modelu) s operátorem objektu charakterizujeme účelovou funkcí $J[S(O, O_m)]$, kde S je tzv. ztrátová funkce, O a O_m operátor objektu, respektive jeho modelu. Podle účelové funkce můžeme porovnávat různé struktury modelů. Výběr účelové funkce výrazně ovlivňuje algoritmus řešení úlohy identifikace.

V prvním kroku pro kvantitativní hodnocení stupně blízkosti operátoru volíme vhodnou ztrátovou funkci $S(O, O_m)$. Blízkost operátorů se posuzuje na bázi odchylek korespondujících signálů nebo parametrů operátorů modelů a objektů. Přestože přímo pozorovatelné jsou především signály, rozeznáváme v zásadě tři způsoby formování odchylek mezi modelem a objektem.

a) Chyba výstupu – vytváříme odchylku mezi výstupy paralelně zapojeného objektu a

modelu. Vektor e zahrnuje měřitelné i neměřitelné poruchy.

Chyba výstupu je:

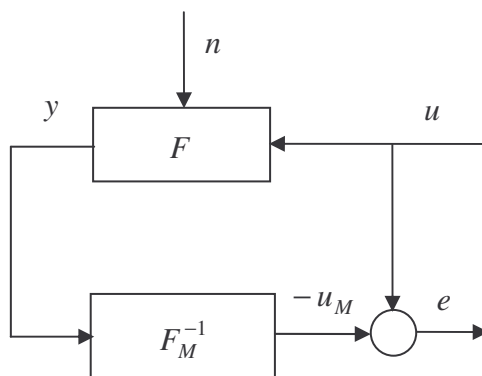


Obrázek 3. Chyba výstupu

b) Chyba vstupu – vytváříme odchylku na straně vstupních signálů modelu a objektu v sériovém zapojení. Chyba je daná jako:

$$e = u - u_m = u - F_m^{-1}(u) \quad (8)$$

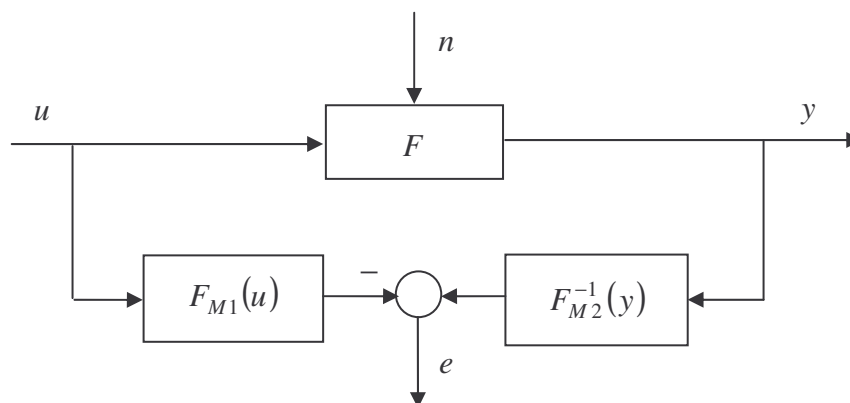
u_m je vstup modelu, generující výstup y . Samozřejmě, F_m^{-1} je inverzní operátor a F_m musí být invertovatelný.



Obrázek 4. Chyba vstupu

c) Chyba rovnice – ve všeobecném případě může být chyba definována jako:

$$e = F_{M2}^{-1}(y) - F_{M1}^{-1}(u) \quad (9)$$



Obrázek 5. Chyba rovnice

F_{M2} představuje invertovatelnou část modelu a celý model je daný jako $F_M = F_{M1}F_{M2}$. Chyba je přímo generovaná ze vstupně-výstupního opisu jako nesoulad mezi levou a pravou stranou rovnice.

Chyba rovnice se většinou používá u modelů ve tvaru diferenciální a diferenční rovnice, přenosových funkcí apod. U některých neparametrických metod, kdy se koeficienty modelu ze souboru dat neurčují metodami odhadu parametrů, není nutno chybu ani kritérium definovat.

2 UŽIVATELSKÝ NÁVOD

Programový systém MATLAB je vysoce výkonný jazyk pro technické výpočty. Integruje výpočty, vizualizace a programování do jednoduše použitelného prostředí, kde problémy i řešení jsou vyjádřeny v přirozeném tvaru. Jde o interaktivní systém jehož základním datovým typem je dvourozměrné pole. Tato vlastnost spolu s množstvím zabudovaných funkcí umožňuje relativně snadné řešení mnoha technických problémů, speciálně takových, které vedou na vektorovou či maticovou formulaci.

Typické oblasti využití jsou:

- 1) Výpočet a odhad
- 2) Vývoj algoritmů
- 3) Modelování, simulace a vývoj prototypů
- 4) Analyzování dat a jejich vizualizace
- 5) Vědecká a inženýrská grafika
- 6) Vývoj aplikací včetně tvorby grafického uživatelského rozhraní

K jeho základním přednostem patří jeho otevřenost, tj. možnost rozšiřování o vlastní funkce, značný počet dostupných, problémově orientovaných balíčků hotových funkcí (tzv. toolboxů), jednoduchá syntaxe a kvalitní implementované algoritmy. Protože lineární teorie automatického řízení využívá v široké míře práce s maticemi, je MATLAB pro použití v této oblasti velice vhodný a stal se světovým standardem. Svědčí o tom velké množství dostupných toolboxů - Control System Toolbox pro klasickou teorii řízení, Signal Processing Toolbox pro syntézu filtrů, System Identification Toolbox pro identifikaci dynamických systémů, Optimization Toolbox pro řešení různých úloh statické optimalizace ap.

Název programu MATLAB napovídá, že jde o "maticovou laboratoř". Je to interaktivní program, ovládaný z příkazového řádku, určený pro odborníky v různých oblastech. Na rozdíl od některých programů pro širokou veřejnost, ovládaných pouze pomocí menu (bez možnosti zadávat příkazy na příkazový řádek), vyžaduje efektivní používání MATLABu určitý stupeň znalostí jeho syntaxe a základního okruhu funkcí. Proto lze vřele doporučit pečlivé seznámení s tímto programem.

Programové prostředí MATLAB, které je nutné k provozu každého toolboxu, je tvořeno vlastním jádrem. Toto jádro (tzv. Kernel) je napsáno ve vyšším programovacím jazyku a posléze zkompileováno. Je v binárním tvaru a je označeno jako MATLAB.EXE. Vlastní jádro obsahuje základní funkce, které se využívají při tvorbě tzv. m-souborů. (m-soubor obsahuje funkci pro specifický účel).

2.1 Popis System Identification Toolboxu

System Identification Toolbox vytváří přesné, zjednodušené modely komplexních systémů ze zašuměných naměřených dat. Dále poskytuje nástroje pro vytváření matematických modelů dynamických systémů na základě pozorování vstupních a výstupních dat. Nabízí flexibilní grafické rozhraní, které napomáhá při vytváření a úpravě dat a modelů. Identifikační metody poskytované tímto toolboxem jsou užitečné pro různé aplikace počínaje návrhem řídicích systémů a zpracováním signálů až do analýzy časových řad.

Funkce System Identification Toolboxu lze rozdělit do následujících oblastí:

- simulace a odhad
- operace s daty
- parametrický odhad
- neparametrický odhad
- tvorba struktury modelu
- operace se strukturou modelu
- konverze modelu
- zobrazení modelu
- extrakce modelu
- kontrola modelu
- odhad neurčitosti modelu
- odhad struktury modelu
- průběžný odhad parametrů

2.2 Operace s daty

2.2.1 LOAD

Funkce **LOAD** slouží pro nahrání hodnot (proměnných) do pracovního prostoru MATLABu. Program MATLAB pracuje se dvěma typy souborů:

- a. binární soubor
- b. ASCII soubor

V binárním souboru jsou data kódována a tím zabírají menší velikostí, ale v jiném programu než MATLAB nelze tyto hodnoty přečíst. Naproti tomu data v ASCII tvaru můžeme zobrazit a upravit v libovolném textovém editoru, nebo načíst do jiného programu. Tato výhoda je ale zaplácena mnohem větší velikostí souboru než v případě binárního kódování.

Binární soubor má příponu **mat** (např. **objekt.mat**). Pokud budeme nahrávat binární soubor, který tuto příponu obsahovat nebude (např. **objekt2.bin**), musíme zadat prepínač **-mat**.

LOAD objekt2.bin -mat

Když za příkaz **LOAD** nenapišeme název souboru, tak se automaticky nahraje soubor pod názvem **matlab.mat**. Toho můžeme využít a pojmenovat tak nejčastěji používaná data. Je povoleno vynechat koncovku **mat** při zápisu jména souboru za příkaz **LOAD**, i když soubor ve skutečnosti tuto příponu má. Např. soubor se bude jmenovat **objekt1.mat** a my můžeme zkrátit zápis na:

LOAD objekt1

Soubor ve formátu ASCII může mít jakoukoliv příponu (např. **objekt3.txt**). Po nahrání dat do pracovního prostoru jsou hodnoty uloženy do proměnných, které byly definovány nahrávaným souborem. Pro výpis proměnných v pracovním prostoru se používá příkaz **WHOS**. Před započítím nové práce je vhodné „vyčistit“ pracovní prostor od předešlých proměnných a to příkazem **CLEAR ALL**.

2.2.2 SAVE

Funkce **SAVE** slouží k uložení hodnot (proměnných) na pevný disk PC. Můžeme ukládat opět do dvou základních typů souboru a to:

- a. binární soubor
- b. ASCII soubor

save “název souboru” – uloží soubor ve tvaru “název souboru”.mat, který je poté možno nahrát použitím funkce **LOAD**.

2.2.3 DETREND

Hodnoty získané měřením na laboratorním nebo průmyslovém zařízení nejsou zpravidla vhodné pro přímé další zpracování. Je nutné hodnoty vhodným způsobem upravit.

Funkce **DETREND** odstraňuje sklon, který mají vstupní a výstupní data. Tato funkce upraví data tak, aby jejich střední hodnota (tj. průměr) byla rovna nule.

Tvar funkce: **zd = DETREND (z)**

Příklad: **load dryer2**

```
z = [y2    u2];
```

```
z = dtrend ( z );
```

Středník na konci řádku zakáže vypsání hodnot na obrazovku. Je samozřejmé, že místo proměnné **zd** můžeme zadat libovolnou jinou a pokud bychom neuvedli proměnou žádnou, výsledek se uloží do proměnné **ans** (z angličtiny **answer** = výsledek). Tato skutečnost všeobecně platí pro většinu funkcí.

2.3 Parametrický odhad

Po nahrání a úpravě dat stojíme před problémem, jaký typ modelu použít pro popis chování identifikovaného objektu. Je nutné zvážit jaký typ dat chceme identifikovat, tj. jestli máme k dispozici hodnoty závislé pouze na čase (angl. **time - series data**) nebo data, která byla buzena určitým vstupním signálem, např. data s několika vstupními signály. V této kapitole si popíšeme modely, které používají k odhadu parametrů buď metodu

nejmenších čtverců (angl. least squares method) nebo metoda instrumentální proměnné (angl. instrumental variables method).

2.3.1 AR

Funkce **AR** je určena pro odhad parametrů modelu **AR** z časově závislých dat. Jsou to data, která byla naměřena na výstupu identifikovaného objektu v závislosti na čase. Funkce **AR** používá k odhadu parametrů metodu nejmenších čtverců.

Tvar funkce: **th = AR(y, nn)**

nn = na

Kde y je sloupcový vektor obsahující časově závislé výstupní data skalár nn specifikuje řád modelu, tj. řád diferenční rovnice.

Výsledek (tedy odhad parametrů) se uloží ve formátu **IDPOLY**. Formát **IDPOLY** je základním formátem System Identification Toolbox a jsou v něm uloženy informace o struktuře modelu, vzorkovacím intervalu **Ts**, chybě predikce, dále odhady parametrů **A, B, C, D, F** (samozřejmě jenom pokud je použitý model obsahuje). Informace obsažené v **IDPOLY** formátu dokonce uchovávají přesné datum (včetně hodiny a minuty), kdy byl model vytvořen.

Vraťme se však k **AR** modelu, který si popíšeme lineární maticovou stochastickou rovnicí (uvažujeme model v diskrétním tvaru):

$$A(z^{-1})y(k) = B(z^{-1})u(k)z^{-nk} \quad (10)$$

kde $A(z^{-1})$ je maticový polynom a $n_s(k)$ je bílý šum. Maticový polynom má tvar:

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \quad (11)$$

Po rozepsání polynomů dostane diferenční rovnici:

$$y(k) + a_1 y(k-1) + \dots + a_{na} y(k-na) = n_s(k) \quad (12)$$

Blíže si model **AR** rozebírat nebudeme, protože model **AR** je pouze jistým zjednodušením modelu **ARX** a ten si podrobně probereme v následujícím článku.

2.3.2 ARX

Funkce **ARX** je určena k odhadu parametrů modelu **ARX** resp. **AR**. Bylo by vhodné vysvětlit, jak model **ARX** získáme.

Diskrétní deterministický objekt s dopravním zpožděním si za použití Z – transformace můžeme popsat vztahem:

$$A(z^{-1})y(k) = B(z^{-1})u(k)z^{-nk} \quad (13)$$

neboli:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) \quad (14)$$

kde $A(z^{-1}), B(z^{-1})$ jsou maticové polynomy, $y(k)$ je vektor výstupních veličin, $u(k)$ vektor vstupních veličin a nk je počet kroků dopravního zpoždění.

Objekt můžeme popsat také maticovou diskretní přenosovou funkcí:

$$F(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} z^{-nk} \quad (15)$$

Každý reálný objekt je ve větší nebo v menší míře ovlivňován náhodnými neměřitelnými vlivy, které označíme $n(k)$. Účinek těchto vlivů u lineárních a linearizovatelných objektů (podle principu superpozice) modelujeme pomocí náhodného vektoru $n(k)$, který připočítáváme obvykle k nezašuměnému výstupu objektu. Potom stochastický objekt si můžeme představit jako objekt deterministický s výstupem $y_d(k)$, ke kterému se přičítá vektor šumového signálu $n(k)$.



Obrázek 6. Model stochastického objektu

Potom obecný lineární časově invariantní stochastický objekt s dopravním zpožděním můžeme popsat vztahy:

$$A(z^{-1})y_d(k) = B(z^{-1})u(k)z^{-nk} \quad (16)$$

$$y(k) = y_d(k) + n(k) \quad (17)$$

Z rovnice (17) vyjádříme $y_d(k)$:

$$y_d(k) = y(k) - n(k) \quad (18)$$

Do vztahu (16) dosadíme za $y_d(k)$ z rovnice (18) a dostaneme:

$$A(z^{-1})y(k) = B(z^{-1})u(k)z^{-nk} + A(z^{-1})n(k) \quad (19)$$

Rovnice (19) se nazývá lineární maticová stochastická rovnice.

Pro zjednodušení budeme dále uvažovat jednorozměrový stochastický objekt bez dopravního zpoždění, který lze popsat rovnicí:

$$A(z^{-1})y(k) = B(z^{-1})u(k) + A(z^{-1})n(k) \quad (20)$$

kde $A(z^{-1}), B(z^{-1})$ jsou maticové polynomy:

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \quad (21)$$

$$B(z^{-1}) = b_1 + b_2 z^{-1} + \dots + b_{nb} z^{-nb+1} \quad (22)$$

Součin $A(z^{-1})n(k)$ se nazývá chybou rovnice $e_r(k)$:

$$e_r(k) = A(z^{-1})n(k) \quad (23)$$

Protože víme, že libovolný náhodný signál $n(k)$ vznikl filtrací bílého šumu $n_s(k)$, dostaneme tvar, který se nazývá autoregrese (AR):

$$\frac{n(k)}{n_s(k)} = \frac{1}{C(z^{-1})} \quad (24)$$

Z rovnice (24) vyjádříme náhodný signál $n(k)$

$$n(k) = \frac{1}{C(z^{-1})} n_s(k) \quad (25)$$

Ale chyba rovnice $e_r(k)$ je také náhodným signálem a potom s použitím vztahu (25) dostaneme:

$$e_r(k) = \frac{1}{C(z^{-1})} n_s(k) \quad (26)$$

Pokud dosadíme rovnici (23) do rovnice (20) můžeme psát:

$$A(z^{-1})y(k) = B(z^{-1})u(k) + e_r(k) \quad (27)$$

Dosazením vztahu (26) do rovnice (27) dostaneme lineární maticovou stochastickou rovnici:

$$A(z^{-1})y(k) = B(z^{-1})u(k) + \frac{1}{C(z^{-1})} n_s(k) \quad (28)$$

Celou rovnici (28) můžeme vynásobit polynomem $C(z^{-1})$:

$$A(z^{-1})C(z^{-1})y(k) = B(z^{-1})C(z^{-1})u(k) + n_s(k) \quad (29)$$

Formálně označíme součiny:

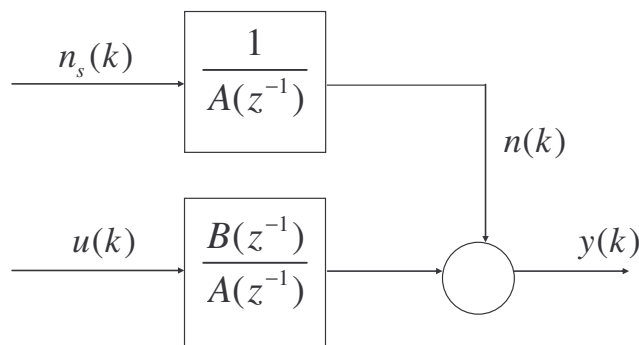
$$A(z^{-1})C(z^{-1}) = A(z^{-1}) \quad (30)$$

$$B(z^{-1})C(z^{-1}) = B(z^{-1}) \quad (31)$$

Touto úpravou jsme docílili toho, že nové polynomy $A(z^{-1})$ a $B(z^{-1})$ už v sobě obsahují polynom $C(z^{-1})$.

Model, který dostaneme se nazývá ARX model a má tvar (pro zjednodušení neuvažujeme dopravní zpoždění nk):

$$A(z^{-1})y(k) = B(z^{-1})u(k) + n_s(k) \quad (32)$$



Obrázek 7. Blokové schéma ARX modelu

Oproti jiným stochastickým modelům má model ARX v polynomech $A(z^{-1})$ a $B(z^{-1})$ zahrnut jak popis dynamického chování objektu, tak popis vlastností šumového signálu.

V System Identification Toolboxu je model ARX naprogramován jako samostatná funkce.

Tvar funkce: **th = ARX (DATA, nn)**

kde:

nn = [na nb nk]

na resp. **nb** udává řád modelu **ARX** (čili řád diferenční rovnice).

nk je dopravní zpoždění.

Výsledek se uloží ve formátu **IDPOLY** do proměnné **M**. Model ARX používá k odhadu parametrů metodu nejmenších čtverců a můžeme si ho popsat lineární maticovou stochastickou rovnicí:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + n_s(k) \quad (33)$$

V této rovnici je na rozdíl od rovnice (32) zahrnuto dopravní zpoždění nk . Po rozepsání polynomů dostaneme:

$$y(k) + a_1 y(k-1) + \dots + a_{na} y(k-na) = b_1 u(k-nk) + b_2 u(k-nk-1) + \dots + b_{nb} u(k-nk-nb+1) + n_s(k) \quad (34)$$

Pokud použijeme k identifikaci mnoharozměrové (více vstupé) signály, přejde rovnice (32) na:

$$A(z^{-1})y(k) = B_1(z^{-1})u_1(k-nk_1) + \dots + B_{nu}(z^{-1})u_{nu}(k-nk_{nu}) + n_s(k) \quad (35)$$

Všechna vstupní data budou popsána maticí $\mathbf{u} = [\mathbf{u1} \ \mathbf{u2} \ \dots \ \mathbf{unu}]$, kde $\mathbf{u1}, \dots, \mathbf{unu}$ budou sloupcové vektory.

Když použijeme k identifikaci časově závislá data, změní se matice $\mathbf{z} = [\mathbf{y} \ \mathbf{u}]$ na vektor $\mathbf{z} = \mathbf{y}$ a tím pádem se redukuje model ARX na model AR:

$$A(z^{-1})y(k) = n_s(k) \quad (36)$$

Všechny uvedené příklady jsou identické a poskytují stejné výsledky. Poslední zápis je sice nejkratší, ale také nejméně přehledný (původní program se zkrátil z 5 řádků na řádky 2) V matici $\mathbf{z}$ se vektory mohou oddělit buď mezerou nebo čárkou. Pokud bychom k oddělení použili středník, nevznikla by matice, ale sloupcový vektor – v našem případě z vektorů y_2 a u_2 .

Příklad 1: load dryer2

```
z = [y2 u2]

z = dtrend (z)

nn = [2 2 3]

th = arx (z,nn)
```

Příklad2: load dryer2

```
z = [y2,u2]

z = dtrend (z)

th = arx (z, [2 2 3])
```

Příklad3: load dryer2

```
th = arx (dtrend ([y2 u2]), [2 2 3])
```

Protože jsme použili (viz. **nn**) **na = 2** (počet parametrů polynomu a bude tedy dva) **nb = 2** (počet parametrů polynomu b také dva) a velikost dopravního zpoždění **nk = 3**, dostaneme podle rovnice (34) konkrétní tvar pro náš případ, tedy:

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = b_1 u(k-3) + b_2 u(k-4) + n_s(k) \quad (37)$$

Abychom zobrazili informace uložené v proměnné **M**, použijeme funkci **PRESENT** s parametrem **th** v závorce. Podrobněji se s touto funkcí seznámíme na jiném místě.

2.3.3 ARMAX

Funkce **ARMAX** odhaduje parametry modelu **ARMAX** resp. ARMA. Ukážeme si, jak model **ARMAX** získáme.

Jednorozměrový stochastický objekt s dopravním zpožděním popíšeme rovnicí:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + A(z^{-1})n(k) \quad (38)$$

nebo:

$$A(z^{-1})y(k) = B(z^{-1})u(k)z^{-nk} + A(z^{-1})n(k) \quad (39)$$

Tyto rovnice se nazývají lineární maticové stochastické rovnice. Jakým způsobem se k nim dostaneme je popsáno vztahy (13) až (19). V rovnici (20) resp. (39) značí $A(z^{-1})$ a $B(z^{-1})$ maticové polynomy:

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \quad (40)$$

$$B(z^{-1}) = b_1 + b_2 z^{-1} + \dots + b_{nb} z^{-nb+1} \quad (41)$$

V rovnici (38) označuje $n(k)$ náhodný signál, nk dopravní zpoždění, $y(k)$ je vektor výstupních veličin a konečně $u(k)$ vektor vstupních hodnot.

Při matematickém popisu vlastností náhodného (šumového) signálu $n(k)$ vycházíme z představy, že libovolný diskretní náhodný signál vznikl filtrací bílého šumu $n_s(k)$. Diskretní přenosovou funkci filtru můžeme napsat ve tvaru:

$$F_f(z^{-1}) = \frac{n(k)}{n_s(k)} = D(z^{-1}) \quad (42)$$

Z rovnice (42) můžeme vyjádřit náhodný $n(k)$:

$$n(k) = D(z^{-1})n_s(k) \quad (43)$$

Vztah (43) se nazývá klouzavý průměr (angl. moving average, z toho zkratka MA).

Přenosovou funkci filtru můžeme napsat také ve tvaru:

$$F_f(z^{-1}) = \frac{n(k)}{n_s(k)} = \frac{1}{C(z^{-1})} \quad (44)$$

Z rovnice (44) vyjádříme $n(k)$:

$$n(k) = \frac{1}{C(z^{-1})} n_s(k) \quad (45)$$

Tvar (44) se nazývá autoregrese (odtud zkratka AR). Pokud zkombinujeme autoregresy AR s klouzavým průměrem MA (42) dostaneme tvar:

$$F_f(z^{-1}) = \frac{n(k)}{n_s(k)} = \frac{D(z^{-1})}{C(z^{-1})} \quad (46)$$

úpravou (46) dostaneme vztah:

$$n(k) = \frac{D(z^{-1})}{C(z^{-1})} n_s(k) \quad (47)$$

Ze vztahu (47) je zřejmé, že náhodnou složku $n(k)$, která ovlivňuje chování objektu, můžeme pokládat za odezvu zvláštní diskrétní soustavy na bílý šum $n_s(k)$.

Dosazením vztahu (47) do rovnice (38) dostaneme lineární maticovou stochastickou rovnici:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + \frac{A(z^{-1})D(z^{-1})}{C(z^{-1})} n_s(k) \quad (48)$$

Součin $A(z^{-1}) n(k)$ v rovnici (38) se nazývá chyba rovnice $e_r(k)$:

$$e_r(k) = A(z^{-1})n(k) \quad (49)$$

Potom můžeme vztah (38) napsat jako:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + e_r(k) \quad (50)$$

jelikož je chyba rovnice $e_r(k)$ také náhodným signálem, dostaneme s použitím vztahu (47) tvar:

$$e_r(k) = n(k) = \frac{D(z^{-1})}{C(z^{-1})} n_s(k) \quad (51)$$

Dosazením (51) do rovnice (50) dostaneme:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + \frac{D(z^{-1})}{C(z^{-1})} n_s(k) \quad (52)$$

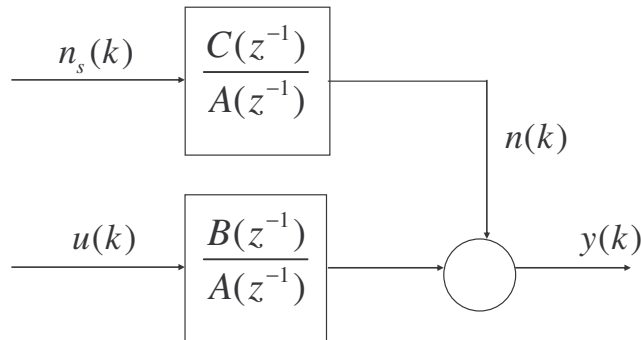
Výraz $\frac{D(z^{-1})}{C(z^{-1})}$ formálně označíme jako $C(z^{-1})$ a potom model, který získáme se nazývá

ARMAX model:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + C(z^{-1})n_s(k) \quad (53)$$

Rovnici (53) můžeme také napsat jako:

$$A(z^{-1})y(k) = B(z^{-1})u(k)z^{-nk} + C(z^{-1})n_s(k) \quad (54)$$



Obrázek 8. Blokové schéma ARMAX modelu

Pokud bychom neuvažovali dopravní zpoždění nk , tak by přešla rovnice (54) na:

$$A(z^{-1})y(k) = B(z^{-1})u(k) + C(z^{-1})n_s(k) \quad (55)$$

Model ARMAX (na rozdíl od modelu ARX, který v polynomech $A(z^{-1})$ a $B(z^{-1})$ popisuje zároveň chování dynamického objektu a vlastnosti působícího šumového signálu) používá k popisu vlastností působícího šumového signálu polynomu $C(z^{-1})$.

Tímto jsme si objasnili tvorbu modelu ARMAX a teď se podíváme, jak je tento model řešen v System Identification Toolboxu.

Tvar funkce: **th = ARMAX (z, nn)**

kde z je matice výstupních a vstupních hodnot $z = [y \ u]$, y je sloupcový vektor výstupních dat, u je sloupcový vektor vstupních dat.

$nn = [na \quad nb \quad nc \quad nk]$ na, nb, nc udává řád modelu ARMAX (čili řád diferenční rovnice), nk je dopravní zpoždění.

Výsledek (odhad) se uloží ve formátu **IDPOLY** do proměnné **th**. Model ARMAX používá k odhadu parametrů chybu predikce (angl. prediction error). Chybu predikce definujeme jako rozdíl mezi skutečnou hodnotou výstupu a předikovanou hodnotou výstupu, vypočtenou z parametrů modelu a z naměřených hodnot.

Chyba predikce má tvar:

$$e(k) = y(k) - y_p(k) \quad (56)$$

kde $y(k)$ je skutečná hodnota výstupu a $y_p(k)$ je predikovaná hodnota výstupu.

Model ARMAX si můžeme popsat lineární maticovou stochastickou rovnicí:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + C(z^{-1})n_s(k) \quad (57)$$

$A(z^{-1})$, $B(z^{-1})$ a $C(z^{-1})$ jsou maticové polynomy.

V případě, že použijeme k identifikaci časově závislé data, změní se matice $\mathbf{z} = [\mathbf{y} \quad \mathbf{u}]$ na sloupcový vektor $\mathbf{z} = \mathbf{y}$, $\mathbf{nn} = [\mathbf{na} \quad \mathbf{nc}]$ a model ARMAX přejde na model ARMA. Model ARMA můžeme popsat maticovou stochastickou rovnicí:

$$A(z^{-1})y(k) = C(z^{-1})n_s(k) \quad (58)$$

2.3.4 IVAR

Funkce **IVAR** odhaduje parametry AR modelu a k tomuto odhadu využívá aproximace optimální volby při použití IV (angl. instrumental variable) metody.

Tvar funkce: **th = IVAR(y, nn)** **nn = na**

kde $\mathbf{y}$ je sloupcový vektor obsahující časově závislá výstupní data, skalár nn specifikuje řád modelu (počet parametrů polynomu a , neboli řád diferenční rovnice).

Výsledek odhadu se uloží ve formátu **IDPOLY** do proměnné **th**. Tato funkce je určena pouze pro časově závislá data a pokud bychom chtěli identifikovat objekt, který je popsán vstupními a výstupními hodnotami, musíme požit funkci **IV4** (ta bude probrána později), která používá k odhadu parametrů také IV metodu.

Funkce **IVAR** na rozdíl od funkce **AR**, která používá k odhadu parametrů metodu nejmenších čtverců, využívá k odhadu IV metodu a není citlivá na barvu šumového signálu.

Strukturu **AR** modelu si popíšeme pomocí lineární maticové stochastické rovnice:

$$A(z^{-1})y(k) = n(k) \quad (59)$$

kde $A(z^{-1})$ je maticový polynom a $n(k)$ je šumový signál.

Maticový polynom má tvar:

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \quad (60)$$

Po rozepsání polynomů dostaneme diferenční rovnici:

$$y(k) + a_1 y(k-1) + \dots + a_{na} y(k-na) = n(k) \quad (61)$$

Příklad: Zkusíme si identifikovat náhodně vygenerované hodnoty. K vytvoření sloupcového vektoru použijeme funkci **RAND** (ta je součástí jádra MATLABu). Funkce má tvar:

RAND (m, n), kde m je počet řádků a n počet sloupců náhodně vygenerovaných hodnot.

$$\mathbf{y} = \text{rand} (100, 1)$$

$$\mathbf{y} = \text{dtrend} (\mathbf{y})$$

$$\mathbf{th} = \text{ivar} (\mathbf{y}, 2)$$

Sloupcový vektor **y** obsahuje 100 hodnot, a protože tyto hodnoty jsou náhodné, uvedeme si pouze obecný tvar výsledku. V příkladu jsme použili ve funkci **IVAR** hodnoty **nn = 2**, a proto dostaneme podle vztahu (60) maticový polynom ve tvaru:

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} \quad (62)$$

resp. diferenční rovnici podle (61):

$$y(k) + a_1 y(k-1) + a_2 y(k-2) = n(k) \quad (63)$$

Hodnoty a_1 , a_2 nám zobrazí funkce **PRESENT**.

2.3.5 IV4

Funkce **IV4** odhaduje parametry ARX modelu a k tomuto odhadu využívá aproximace čtyř etapové IV (angl. instrumental variable) metody.

Tvar funkce: **th = IV4 (z, n)**

kde **z** je matice výstupních a vstupních hodnot **z = [y u]**, **y** je sloupcový vektor výstupních dat, **u** je sloupcový vektor vstupních dat.

nn = [na nb nk] **na** resp. **nb** udává řád modelu ARX (počet parametrů a, resp. b – tj. řád diferenční rovnice), **nk** je dopravní zpoždění.

Výsledek se uloží ve formátu **IDPOLY** do proměnné **th**, z které si ho můžeme vhodnými funkcemi zobrazit.

Funkce **IV4** se liší od funkce **ARX** tím, že k odhadu parametrů modelu ARX nepoužívá metodu nejmenších čtverců ale metodu IV a na rozdíl od funkce **ARX** není citlivá na barvu šumového signálu.

V případě, kdy $\mathbf{nn} = [\mathbf{na} \quad \mathbf{nb} \quad \mathbf{nk}]$ si model ARX popíšeme lineární maticovou maticovou stochastickou rovnicí:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + n(k) \quad (64)$$

nebo:

$$A(z^{-1})y(k) = B(z^{-1})u(k)z^{-nk} + n(k) \quad (65)$$

kde nk je dopravní zpoždění a $n(k)$ je šumový signál.

Odvození modelu ARX bylo popsáno u funkce **ARX** rovnicemi (49) až (32) – viz. kapitola 2.3.2.

Pokud je dopravní zpoždění nk nulové a tedy $\mathbf{nn} = [\mathbf{na} \quad \mathbf{nb}]$, redukuje se tvar (64) resp. (65) na:

$$A(z^{-1})y(k) = B(z^{-1})u(k) + n(k) \quad (66)$$

kde $n(k)$ je šumový signál, $A(z^{-1})$ a $B(z^{-1})$ jsou maticové polynomy:

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \quad (67)$$

$$B(z^{-1}) = b_1 + b_2 z^{-1} + \dots + b_{nb} z^{-nb+1} \quad (68)$$

Po rozepsání polynomů v rovnici (64) resp. (65) obdržíme diferenční rovnici:

$$y(k) + a_1 y(k-1) + \dots + a_{na} y(k-na) = b_1 u(k-nk) + b_2 u(k-nk-1) + \dots + b_{nb} u(k-nk-nb+1) + n(k) \quad (69)$$

nebo při použití rovnice (66), kde je dopravní zpoždění nulové dostaneme:

$$y(k) + a_1 y(k-1) + \dots + a_{na} y(k-na) = b_1 u(k) + b_2 u(k-1) + \dots + b_{nb} u(k-nb) + n(k) \quad (70)$$

Pokud použijeme k identifikaci mnoharozměrové (více vstupé) hodnoty, změní se sloupcový vektor $\mathbf{u}$ na matici $\mathbf{u} = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \dots \quad \mathbf{u}_{nu}]$ resp. matice $\mathbf{z}$ na $\mathbf{z} = [\mathbf{y} \quad \mathbf{u}_1 \quad \mathbf{u}_2 \dots \mathbf{u}_{nu}]$ a rovnice (64) nabude tvaru:

$$A(z^{-1})y(k) = B_1(z^{-1})u_1(k-nk_1) + \dots + B_{nu}(z^{-1})u_{nu}(k-nk_{nu}) + n(k) \quad (71)$$

2.3.6 OE

Funkce **OE** odhaduje parametry modelu výstupní chyby (anglicky output – error model).

Tvar funkce: $\mathbf{th} = \mathbf{OE}(\mathbf{z}, \mathbf{nn})$

kde $\mathbf{z}$ je matice výstupních a vstupních hodnot $\mathbf{z} = [\mathbf{y} \ \mathbf{u}]$, $\mathbf{y}$ je sloupcový vektor výstupních dat, $\mathbf{u}$ je sloupcový vektor vstupních dat.

$\mathbf{nn} = [\mathbf{nb} \ \mathbf{nf} \ \mathbf{nk}]$ $\mathbf{nb}$ resp. $\mathbf{nf}$ udává řád modelu OE (modelu výstupní chyby) – udává řád diferenční rovnice, tj. počet parametrů b resp. f , $\mathbf{nk}$ je dopravní zpoždění.

Výsledek (odhad parametrů) se uloží ve formátu **IDPOLY** (to je základní formát System Identification Toolbox – jsou v něm uloženy informace o struktuře modelu, vzorkovacím intervalu T atd. do proměnné **th**).

Funkce **OE** využívá k odhadu parametrů tohoto modelu chybu predikce. Chybu predikce definujeme jako rozdíl mezi skutečnou hodnotou výstupu a předikovanou hodnotou výstupu, vypočtenou z parametrů modelu a z naměřených hodnot.

Chyba predikce má tvar:

$$e(k) = y(k) - y_p(k) \quad (72)$$

kde $y(k)$ je skutečná hodnota výstupu a $y_p(k)$ je predikovaná hodnota výstupu.

Model výstupní chyby (model OE) můžeme popsat lineární maticovou stochastickou rovnicí:

$$y(k) = \frac{B(z^{-1})}{F(z^{-1})} u(k) z^{-nk} + n_s(k) \quad (73)$$

neboli:

$$y(k) = \frac{B(z^{-1})}{F(z^{-1})} u(k - nk) + n_s(k) \quad (74)$$

kde $n_s(k)$ je bílý šum. $B(z^{-1})$ a $F(z^{-1})$ jsou maticové polynomy tvaru:

$$\begin{aligned} F(z^{-1}) &= 1 + f_1 z^{-1} + \dots + f_{nf} z^{-nf} \\ B(z^{-1}) &= b_1 + b_2 z^{-1} + \dots + b_{nb} z^{-nb+1} \end{aligned} \quad (75)$$

Funkce **OE** pracuje pouze s jednorozměrovými signály. Pro obecný případ je k dispozici funkce **PEM**, která bude probírána později.

2.3.7 BJ

Funkce **BJ** odhaduje parametry modelu Box – Jenkins.

Tvar funkce: **th = BJ(z, nn)**

kde $\mathbf{z}$ je matice výstupních a vstupních hodnot $\mathbf{z} = [\mathbf{y} \ \mathbf{u}]$, $\mathbf{y}$ je sloupcový vektor výstupních dat, $\mathbf{u}$ je sloupcový vektor vstupních dat.

$\mathbf{nn} = [\mathbf{nb} \quad \mathbf{nc} \quad \mathbf{nd} \quad \mathbf{nf} \quad \mathbf{nk}]$ $\mathbf{nb}$, $\mathbf{nc}$, $\mathbf{nd}$, $\mathbf{nf}$ udává řád modelu Box – Jenkins (tj. řád diferenční rovnice), $\mathbf{nk}$ je dopravní zpoždění.

Funkce **BJ** používá k odhadu parametrů modelu chybu predikce (angl. prediction error). Je to rozdíl mezi skutečnou hodnotou výstupu $y(k)$ a předikovanou hodnotou výstupu $y_p(k)$.

Chyba predikce má tvar:

$$e(k) = y(k) - y_p(k) \quad (76)$$

Model Box-Jenkins lze popsat lineární maticovou stochastickou rovnicí:

$$y(k) = \frac{B(z^{-1})}{F(z^{-1})} u(k - nk) + n_s(k) \quad (77)$$

neboli:

$$y(k) = \frac{B(z^{-1})}{F(z^{-1})} u(k) z^{-nk} + \frac{C(z^{-1})}{D(z^{-1})} n_s(k) \quad (78)$$

Model Box – Jenkins na rozdíl od modelu **OE** (angl. output error model – model výstupní chyby) popisuje vlastnosti šumového signálu pomocí polynomů $C(z^{-1})$ a $D(z^{-1})$.

Pokud bude $\mathbf{nc} = \mathbf{0}$ a $\mathbf{nd} = \mathbf{0}$ dostaneme pro řádkový vektor $\mathbf{nn}$ tvar: $\mathbf{nn} = [\mathbf{nb} \quad \mathbf{nf} \quad \mathbf{nk}]$ a model Box – Jenkins přejde na model **OE**, protože lineární maticová stochastická rovnice se (75) změní na:

$$y(k) = \frac{B(z^{-1})}{F(z^{-1})} u(k - nk) + n_s(k) \quad (79)$$

Funkce **BJ** je určena pouze pro jednorozměrové vstupní signály (pro obecný tvar signálu je určena funkce **PEM**).

2.3.8 PEM

Funkce **PEM** odhaduje parametry obecného lineárního modelu.

Tvar funkce: $\mathbf{th} = \mathbf{PEM}(\mathbf{z}, \mathbf{nn})$

kde $\mathbf{z}$ je matice výstupních a vstupních hodnot $\mathbf{z} = [\mathbf{y} \quad \mathbf{u}]$, $\mathbf{y}$ je sloupcový vektor výstupních dat, $\mathbf{u}$ je sloupcový vektor vstupních dat.

$\mathbf{nn} = [\mathbf{na} \quad \mathbf{nb} \quad \mathbf{nc} \quad \mathbf{nd} \quad \mathbf{nf} \quad \mathbf{nk}]$ $\mathbf{na}$, $\mathbf{nb}$, $\mathbf{nc}$, $\mathbf{nd}$, $\mathbf{nf}$ -udává řád modelu (tj.řád diferenční rovnice), $\mathbf{nk}$ je dopravní zpoždění.

Obecný lineární model můžeme popsat maticovou stochastickou rovnicí:

$$A(z^{-1})y(k) = \frac{B(z^{-1})}{F(z^{-1})}u(k - nk) + \frac{C(z^{-1})}{D(z^{-1})}n_s(k) \quad (80)$$

kde $n_s(k)$ je bílý šum a $A(z^{-1})$, $B(z^{-1})$, $C(z^{-1})$, $D(z^{-1})$, $F(z^{-1})$ jsou maticové polynomy.

Z obecného lineárního modelu lze získat tyto typy modelů:

- 1) ARX model – získáme ho tehdy, pokud bude platit $\mathbf{nc} = \mathbf{nd} = \mathbf{nf} = \mathbf{0}$ a tedy

$\mathbf{nn} = [\mathbf{na} \quad \mathbf{nb} \quad \mathbf{nk}]$. Lineární maticová stochastická rovnice bude mít tvar:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + n_s(k) \quad (81)$$

- 2) ARMAX model – ten dostaneme pro $\mathbf{nd} = \mathbf{nf} = \mathbf{0}$, $\mathbf{nn} = [\mathbf{na} \quad \mathbf{nb} \quad \mathbf{nc} \quad \mathbf{nk}]$ a z rovnice obecného lineárního modelu dostaneme:

$$A(z^{-1})y(k) = B(z^{-1})u(k - nk) + C(z^{-1})n_s(k) \quad (82)$$

- 3) OE model – neboli model výstupní chyby (angl. output error model) obdržíme pro:

$\mathbf{na} = \mathbf{nc} = \mathbf{nd} = \mathbf{0}$ a tedy $\mathbf{nn} = [\mathbf{nb} \quad \mathbf{nf} \quad \mathbf{nk}]$.

Lineární maticová stochastická rovnice dostane tvar:

$$y(k) = \frac{B(z^{-1})}{F(z^{-1})}u(k - nk) + n_s(k) \quad (83)$$

- 4) BJ model – neboli Box – Jenkins model dostaneme pro $\mathbf{na} = \mathbf{0}$, pak

$\mathbf{nn} = [\mathbf{nb} \quad \mathbf{nc} \quad \mathbf{nd} \quad \mathbf{nf} \quad \mathbf{nk}]$ a z rovnice obecného lineárního modelu vznikne:

$$: \quad y(k) = \frac{B(z^{-1})}{F(z^{-1})}u(k - nk) + \frac{C(z^{-1})}{D(z^{-1})}n_s(k) \quad (84)$$

2.4 Neparametrický odhad

V kapitole 2.3 byly popsány funkce, které poskytovaly parametrické modely. Typickým představitelem parametrických modelů je např. diferenční rovnice:

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = b_1 u(k-1) + \dots + b_m u(k-m) \quad (85)$$

Výhodou parametrických modelů je jejich snadné modelování, protože pomocí malého objemu údajů o parametrech systému popíšeme jeho dynamické chování.

V této části se budeme zabývat funkcemi popisujícími neparametrické modely. Neparametrické modely vyjadřují relaci mezi závisle a nezávisle proměnnou, nevyžadují žádné informace o struktuře systému, ale jejich modelování je za použití výpočetní techniky těžkopádné a náročné na kapacitu paměti. Typickým představitelem neparametrických modelů je model ve tvaru impulsní nebo frekvenční charakteristiky.

2.4.1 CRA

Funkce **CRA** pomocí korelační analýzy odhaduje impulsní funkci. Korelační analýza patří mezi statistické a pravděpodobnostní metody identifikace. Tyto metody narozdíl od metod deterministických, které používají testovací vstupní signály, při kterých musí být regulační obvod rozpojen, nevyžadují aktivní buzení objektu, ale využívají pouze údaje získané z provozního měření. Případně na buzení objektu používají speciální náhodné testovací signály přidávané k provozním signálům. Nevýhodou statických metod je větší objem náročnějších výpočtů vyžadujících použití výpočetní techniky.

Korelační metody jsou založeny na momentových charakteristikách náhodných procesů a využívají náhodné poruchy a šумы na stanovení vazeb (tzv. korelací) mezi vstupní a výstupními veličinami objektu.

Nyní si ukážeme, jakým způsobem pomocí korelační analýzy dospějeme k impulsní funkci. Při vyšetřování náhodných procesů v lineárních řídicích obvodech nás zajímá, jak se změní statistické charakteristiky náhodných signálů při průchodu řízenou soustavou.

Je-li vstupní veličina náhodná, ergodická a stacionární, bude výstupní veličina také náhodná, ergodická a stacionární, její statistické charakteristiky však budou jiné. Pro popis vstupních a výstupních signálů použijeme tzv. korelační funkci.

Autokorelační funkce $R_{uu}(\tau)$ udává míru korelace (závislosti) mezi veličinou $u(t)$ v okamžiku t a tou samou veličinou $u(t + \tau)$ o čas τ později

$$R_{uu}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T u(t)u(t + \tau)dt \quad (86)$$

Vzájemná korelační funkce R_{uy} vyjadřuje míru závislosti mezi veličinou $u(t)$ v okamžiku t veličinou $y(t + \tau)$ o čas τ později.

$$R_{uy}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T u(t)y(t+\tau)dt \quad (87)$$

Mezi autokorelační a vzájemnou korelační funkcí platí vztah, který se nazývá Wiener-Hopfova rovnice.

$$R_{uy}(\tau) = \int_0^{\infty} u(t)R_{uu}(\tau-t)dtg(t) \quad (88)$$

Tato integrální rovnice nám dává možnost určit při znalosti obou korelačních funkcí $R_{uu}(\tau)$, $R_{uy}(\tau)$ neznámou impulsní funkci $g(t)$. Řešení se provádí většinou přibližnými způsoby na počítači. V těchto případech horní hranici integrálu Wiener-Hopfovy rovnice zaměníme za konečnou a integraci nahradíme sumací:

$$R_{uy}(\tau) \cong \sum_{i=0}^{N-1} g(iT)R_{uu}(\tau-iT) \quad (89)$$

Zvolený integrál je rozdělen na $N-1$ dílů o délce T . Potom pro diskrétní hodnoty $\tau = 0, T, 2T, \dots, (N-1)T$ dostáváme systém N nezávislých lineárních rovnic pro určení N bodů impulsní funkce.

System Identification Toolbox provádí odhad impulsní funkce příkaz **CRA**.

Tvar funkce: **ir = CRA (z, M, na, plot)**

Kde **z** je matice výstupních a vstupních hodnot **z = [y u]**, **y** je sloupcový vektor výstupních dat, **z** je sloupcový vektor vstupních dat.

M udává interval, pro který se má spočítat odhad impulsní funkce (implicitně je nastaveno **M = 20**).

na popisuje řád (implicitně **na = 10**), pokud bude **na = 0** data nebudou upraveny a korelační funkce se vypočítá z originálních hodnot.

plot – pokud bude : **plot = 0** potlačí se grafické zobrazení odhadu impulsní funkce.

plot = 1 (nastaveno implicitně) vykreslí se odhad impulsní funkce.

plot = 2 zobrazí se autokorelační a vzájemná korelační funkce společně s odhadem impulsní funkce.

Funkce **CRA** pracuje pouze s jednorozměrovými vstupními a výstupními signály.

Příklad: **load iddata1**

z = dtrend (z1);

ir = cra (z, 10, 10, 2);

V příkladu je použito souboru **iddata1.mat** (součást System Identification Toolbox), který obsahuje výstupní a vstupní hodnoty v proměnné **z1**. Za parametr **M** je dosazena hodnota 10 tzn., že odhad impulsní funkce se bude počítat na intervalu -10 až 10.

2.4.2 ETFE

Funkce **ETFE** pomocí přímé Fourierovy transformace odhaduje frekvenční přenos resp. periodogram.

Tvar funkce: **g = ETFE (z)**

Kde **z** je matice výstupních a vstupních hodnot **z = [y u]**, **y** je sloupcový vektor výstupních dat, **u** je sloupcový vektor vstupních dat.

Výsledek odhadu se uloží ve formátu **IDFRD** do proměnné **g**, odkud se může vhodnými funkcemi zobrazit (tyto funkce budou probrány později).

Funkce **ETFE** je určena pouze pro jednorozměrové vstupní a výstupní signály. V případě, že použijeme časově závislá data (tzn. že **z = y**), tak se do proměnné **g** uloží periodogram **y**.

Blíže se funkcí **ETFE** zabývat nebudeme a příklad si uvedeme v následující kapitole popisující funkce **SPA**, která stejně jako funkce **ETFE** odhaduje frekvenční přenos, ovšem za použití spektrální analýzy.

2.4.3 SPA

Funkce **SPA** odhaduje pomocí spektrální analýzy frekvenční přenos

Náhodný signál můžeme kromě autokorelační funkce charakterizovat také spektrální výkonovou hustotou:

$$S_{uu}(\omega) = \int_{-\infty}^{\infty} R_{uu}(\tau) e^{-j\omega\tau} d\tau \quad (90)$$

Spektrální hustotu stacionárního náhodného procesu získáme z korelační funkce pomocí Fourierovy transformace. Tvar (90) se nazývá vlastní spektrální hustota a je vždy reálná.

Vzájemná výkonová spektrální hustota je definována vztahem:

$$S_{uy}(j\omega) = \int_{-\infty}^{\infty} R_{uy}(\tau) e^{-j\omega\tau} d\tau \quad (91)$$

a je obecně komplexní.

Za použití výkonových spektrálních hustot $S_{uu}(\omega)$ a $S_{uy}(j\omega)$ můžeme napsat frekvenční přenos:

$$F(j\omega) = \frac{S_{uy}(j\omega)}{S_{uu}(\omega)} \quad (92)$$

Tento postup se označuje jako spektrální analýza a v System Identification Toolbox je popsán funkcí **SPA**.

Tvar funkce: **g = SPA (z)**

kde **z** je matice výstupních a vstupních hodnot **z = [y u]**, **y** je sloupcový vektor výstupových dat, **u** je sloupcový vektor vstupních dat.

Výsledek odhadu se uloží ve formátu **IDRFD** proměnné **g**, odkud se může vhodnými funkcemi zobrazit.

Funkce **SPA** narozdíl od funkce **ETFE** (která také slouží k odhadu frekvenčního přenosu) může pracovat s mnohorozměrovými hodnotami a potom matice **z** dostane tvar:

$$\mathbf{z} = [\mathbf{y1} \ \mathbf{y2} \dots \mathbf{u1} \ \mathbf{u2} \dots]$$

Příklad: V následujícím příkladu porovnáme výsledky funkcí **ETFE** a **SPA**, které používají k odhadu frekvenčního přenosu $F(j\omega)$ každá jinou metodu.

```
load dryer2
```

```
z = dtrend ([y2 u2]);
```

```
ge = etfe ( z );
```

```
gs = spa ( z );
```

```
bode (ge, gs)
```

Výsledek je zobrazen pomocí amplitudové frekvenční charakteristiky (k tomu je použito funkce **bode**).

Pojmem struktura modelu se označuje jednak řád diferenční rovnice a dále velikost dopravního zpoždění. Při odhadu řádu diferenční rovnice se většinou postupuje tak, že se

výpočet opakuje pro zvyšující se řád a testuje se, zda se tím zlepší přesnost modelu podle zvoleného kritéria.

2.5 Odhad struktury modelu

Pojmem struktura modelu se označuje jednak řád diferenční rovnice a dále velikost dopravního zpoždění. Při odhadu řádu diferenční rovnice se většinou postupuje tak, že se výpočet opakuje pro zvyšující se řád a testuje se, zda se tím zlepší přesnost modelu podle zvoleného kritéria.

V kapitole 2.3, která se zabývá parametrickými modely, byl popsán řádkový vektor **nn**. Právě tento vektor udává řád modelu (tj. řád diferenční rovnice) a velikost dopravního zpoždění. Naskytá se otázka, jakým způsobem tyto parametry navrhnout. Jedním ze způsobů je zkusmé zadání několika hodnot, se kterými se model vypočte, následuje zadání hodnot nových, opět se model spočítá a tento postup se opakuje do té doby, než je model co nejvíce totožný s identifikovaným objektem. Tento postup je zdoluhavý, protože parametrů ve vektoru **nn** může být více, a proto v této kapitole popíšeme funkce, které odhad struktury modelu maximálně zjednodušují.

2.5.1 STRUC

Funkce **STRUC** vytváří maticovou strukturu modelu.

Tvar funkce: **nn = STRUC (na, nb, nk)**

kde parametry **na**, **nb** udávají řád modelu (tj. řád diferenční rovnice) a **nk** popisuje dopravní zpoždění.

Funkce **STRUC** pracuje pouze s jednorozměrovými vstupními signály. Vzhledem k tomu, že proměnná **nn** obsahuje parametry **na**, **nb**, **nk** je funkce **STRUC** určena pouze pro modely typu **ARX** (funkce **ARX** a funkce **IV4**).

2.5.2 ARXSTRUC

Funkce **ARXSTRUC** počítá ztrátovou funkci pro nastavení struktury modelu **ARX** (ten je popsán funkcí **ARX**, která používá k odhadu parametrů metodu nejmenších čtverců).

Tvar funkce: **v = ARXSTRUC (ze, zv, nn)**

kde **nn** představuje matici, která je definována funkcí **STRUC** (viz. kapitola 2.5.1).

Parametr **ze** je matice obsahující výstupní a vstupní hodnoty **ze = [yu]**, ze kterých se bude počítat ztrátová funkce a **zv** je matice, podle které se bude výsledek kontrolovat. Matice **ze** a **zv** může mít každá jiný počet hodnot a pokud bude platit **ze = zv** bude výpočet probíhat rychleji.

2.5.3 IVSTRUC

Funkce **IVSTRUC** slouží pro nastavení struktury modelu **ARX**, který je popsán funkcí **IV4** (ta používá pro odhad parametrů modelu **IV** metodu).

Tvar funkce: **v = IVSTRUC (ze, zv, nn)**

kde **nn** představuje matici, která je definována funkcí **STRUC** (viz kapitola 2.5.1).

Parametr **ze** je matice obsahující výstupní a vstupní hodnoty **ze = [yu]**, ze kterých se bude počítat odhad a **zv** je matice, podle jejíž hodnot bude odhadnutý model simulován.

Výsledky se uloží do matice **v**, ze které budou pomocí funkce **SELSTRUC** analyzovány.

Funkce **IVSTRUC** je určena pouze pro jednorozměrové výstupní systémy.

2.5.4 SELSTRUC

Funkce **SELSTRUC** vybírá řád (strukturu) modelu.

Tvar funkce: **v = SELSTRUC (v)**

nebo: **v = SELSTRUC (v, 0)**

kde **v** je matice získaná jako výstup funkcí **ARXSTRUC** resp. **IVSTRUC**.

2.5.5 Sett

Funkce **SETT** slouží pro přímé nastavení periody vzorkování. Ačkoliv se jedná o funkci, která již zastaralá (místo toho lze použít vlastnost **Ts**).

Tvar funkce: **thn = SETT (th, T)**

kde parametr **th** představuje výstup funkcí buď pro parametrický odhad (ve starším formátu **theta**, proměnná **th**) nebo spektrální analýzu (formát **freqfunc**). **T** je perioda vzorkování.

thn je nový model v **theta** formátu se vzorkovacím intervalem **T**.

Ve skutečnosti použití funkce **SETT** není nutné, protože periodu vzorkování **T** lze nastavit jako jeden z parametrů u všech funkcí probraných v kapitolách 2.3, 2.4.2 a 2.4.3. při popisu těchto funkcí byly tyto parametry záměrně vynechány, protože jejich použití není většinou bezpodmínečně nutné a zbytečně komplikují zápis funkce. A právě pro snadné nastavení periody vzorkování slouží funkce **SETT**.

2.6 Kontrola modelu

2.6.1 IDSIM

Funkce **IDSIM** simuluje systém pomocí odhadnutých parametrů, které jsou v **IDPOLY** formátu.

Tvar funkce: **ys = IDSIM ([ue], model)**

nebo: **ys = IDSIM (u, model)**

kde **u** je sloupcový vektor vstupních dat, **e** představuje náhodný signál (šum) a **model** je odhad parametrů modelu zadaných například v **IDPOLY** formátu.

Pokud bude parametr **e** chybět, dostaneme simulaci neovlivněnou šumem. Simulovaný průběh podle naměřených vstupních hodnot **u** a podle odhadnutých parametrů modelu **IDPOLY** se uloží jako výstup modelu do proměnné **ys**, která je sloupcovým vektorem.

Příklad: **load dryer2**

```
y = dtrend ( y2 );  
u = dtrend ( u2 );  
th = arx ([y      u], [2    2    3]);  
ys = idsim (u, model);  
plot ([y      ys])
```

2.6.2 Compare

Funkce **COMPARE** graficky porovnává odměřený výstup s modelovaným výstupem.

Tvar funkce: **COMPARE (DATA, SYS, M)**

kde **data** je matice výstupních a vstupních hodnot $\mathbf{z} = [\mathbf{y} \quad \mathbf{u}]$, $\mathbf{y}$ je sloupcový vektor výstupních dat, $\mathbf{u}$ je sloupcový vektor vstupních dat a **sys** obsahuje odhadnuté parametry modelu ve formátu **IDPOLY**. **M** určuje horizont predikce (standart **M** = inf).

2.7 Zobrazení modelu

2.7.1 BODEPLOT

Funkce **BODEPLOT** graficky zobrazí amplitudovou a fázovou frekvenční charakteristiku (jedná se o zastaralou funkci, místo ní lze použít funkci **BODE**).

Tvar funkce: **BODEPLOT** (**g**, **C**, **mode**)

Nebo: **BODEPLOT** ([**g1 g2 ... gn**])

kde **g** je výstupní funkcí **TH2FF** nebo **SPA** resp. **ETFE**.

Funkce **BODEPLOT** zobrazuje frekvence v rad/ s.

2.7.2 FFPLLOT

Funkce **FFPLOT** graficky zobrazí amplitudovou a fázovou frekvenční charakteristiku.

Tvar funkce: **FFPLOT** (**g**, **C**, **mode**)

nebo: **FFPLOT** ([**g1 g2 ... gn**])

Parametry mají stejný význam jako u funkce **BODEPLOT** (viz. kapitola 2.7.1) a jediný rozdíl mezi oběma funkcemi je v tom, že funkce **FFPLOT** zobrazuje frekvence v Hz.

2.7.3 IDPLOT

Funkce **IDPLOT** graficky zobrazí vstupní a výstupní hodnoty.

Tvar funkce: **IDPLOT** (**z**, **int**, **T**, **ny**)

kde matice výstupních a vstupních hodnot $\mathbf{z} = [\mathbf{y} \quad \mathbf{u}]$, $\mathbf{y}$ je sloupcový vektor výstupních dat, $\mathbf{u}$ je sloupcový vektor vstupních dat.

int – udává interval, který se má zobrazit

T – perioda vzorkování

ny – počet výstupů

Pokud budeme zobrazovat mnohazměrové signály, zobrazí se vždy jen jeden výstupní a jeden vstupní signál a k zobrazení dalších dojde až po stisku klávesy ENTER.

Příklad: **load iddata1**

idplot (z1, 1... 50, 10)

Zobrazí se prvních 50 výstupních a vstupních hodnot s periodou vzorkování 10.

2.7.4 PRESENT

Funkce **PRESENT** zobrazí informace uložené v **IDPOLY** formátu.

Tvar funkce: **PRESENT (IDDATA)**

Proměnná **IDDATA** obsahuje informace o struktuře modelu a funkce **PRESENT** je dokáže přehledně vypsát. Dále zobrazí ztrátovou funkci, chybu predikce a periodu vzorkování **T**.

2.7.5 ZPLOT

Funkce **ZPLOT** graficky zobrazí polohu pólů a nul.

Tvar funkce : **ZPLOT (zp)**

Nebo: **ZPLOT (ZPFORM (zp1, ...zp5))**

Příklad: **load iddata2**

```
th1 = arx (z2, [2 1 1]);  
th2 = arx (z2, [2 2 2]);  
zp1 = th2zp ( th1 );  
zp2 = th2zp ( th2 );  
zpplot (zpform (zp1, zp2))
```

2.8 Konverze modelu

2.8.1 C2D

Funkce **C2D** konvertuje spojitý model na model diskrétní.

Tvar funkce: **C2D (thc, T)**

kde **thc** je spojitý model popsáný **IDPOLY** formátem a **T** je perioda vzorkování.

Pokud budeme identifikovat naměřené hodnoty, tak samozřejmě System Identification Toolbox odhaduje parametry diskrétního modelu.

2.8.2 D2C

Funkce **C2D** konvertuje diskrétní model na model spojitý.

Tvar funkce: **D2C (thd)**

kde **thd** je diskrétní model popsáný **IDPOLY** formátem.

2.8.3 TH2FF

Funkce **TH2FF** počítá frekvenční přenos z **IDPOLY** formátu. A ačkoliv tato funkce zastaralá, lze místo ní použít funkci **IDFRD**.

Tvar funkce: **G = TH2FF (th)**

kde **th** je výstup funkcí definovaný model jako **IDPOLY**. Výsledek se uloží do proměnné **g** ve formátu **freqfunc**, odkud se může pomocí funkce **BODEPLOT** nebo **FFPLOT** resp. **NYQPLOT** graficky zobrazit.

Kromě frekvenčního přenosu počítá funkce **TH2FF** také standardní odchylku. Pokud bychom chtěli spočítat jen frekvenční přenos, je výhodnější použít funkci **TRF**, která je rychlejší než funkce **TH2FF**.

2.8.4 ZPKDATA

Funkce **ZPKDATA** počítá póly a nuly z **IDPOLY** formátu.

Tvar funkce: **[Z, P, K, TS] = ZPKDATA (MODEL)**

kde **model** je ve formátu **IDPOLY**. Výsledek se uloží do proměnných **z, p, k, ts**.

ZÁVĚR

Primárním cílem této bakalářské práce bylo vytvořit elektronický manuál pro System Identification Toolbox a to takovým způsobem, aby praktické využití jeho funkcí nebylo žádným problémem ani pro člověka, který se v této oblasti nikdy dříve nepohyboval.

V teoretické části je vysvětlen pojem identifikace, rozdíl mezi analytickou a experimentální identifikací, přičemž experimentální identifikaci je věnována větší pozornost. Uvedeno je hlavní rozdělení System Identification Toolbox na jednotlivé části (simulace a odhad, operace s daty, parametrický odhad, neparametrický odhad, tvorba struktury modelu, operace se strukturou modelu, konverze modelu, zobrazení modelu, extrakce modelu, kontrola modelu, odhad neurčitosti modelu, odhad struktury modelu, průběžný odhad parametrů). Ke každému bodu jsou uvedeny funkce, které obsahuje, jejich tvar a možnosti použití.

Dále jsou popsány způsoby vyhodnocení naměřených hodnot a druhy modelů včetně použití či jejich výhod a nevýhod.

SEZNAM POUŽITÝCH ZDROJŮ

- 1) BOBÁL, Vladimír. *Identifikace systémů*. Zlín : Ediční středisko VUT Brno, 1990. 184 s. ISBN 80-214-0125-7
- 2) *Documentation for Mathworks Products - System Identification Toolbox* [online]. c1994-2005 [cit. 2005-02-09]. Dostupný z WWW: <http://www.mathworks.com/access/helpdesk/help/toolbox/ident/ident.shtml>
- 3) FIKAR, Miroslav, MIKLEŠ, Ján. *Identifikácia systémov*. Bratislava : STU v Bratislavě, 1999. 110 s. ISBN 80-227-1177-2.
- 4) NOSKIEVIČ, Petr. *Modelování a identifikace systémů*. Ostrava : Montanex a.s., 1999. 280 s. ISBN 80-7225-030-2.
- 5) LJUNG, Lennart. *System Identification : Theory for the User*. London : Prentice Hall, 1987. 519 s. ISBN 978013881640
- 6) LJUNG, Lennart. *System Identification Toolbox : For Use with MATLAB*. [online] [s.l.] : [s.n.], c2004. 407 s. Dostupný z WWW: http://www.mathworks.com/access/helpdesk/help/pdf_doc/ident/ident.pdf

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

u	vstupní signál
y	výstupní signál
T	perioda vzorkování [s]
$E(u)$	střední hodnota
nk	počet kroků dopravního zpoždění
$g(t)$	impulsní funkce
$e(k)$	chyba predikce
$e_r(k)$	chyba rovnice
$n(k)$	náhodný signál
$n_s(k)$	bílý šum
$F(j\omega)$	frekvenční přenos
$R_{uu}(\tau)$	autokorelační funkce
$R_{uy}(\tau)$	vzájemná korelační funkce
$S_{uu}(\omega)$	spektrální výkonová hustota
$F(z)$	Z-přenos
na	stupeň polynomu $A(z^{-1})$
nb	stupeň polynomu $B(z^{-1})$
nc	stupeň polynomu $C(z^{-1})$
nd	stupeň polynomu $D(z^{-1})$
nf	stupeň polynomu $F(z^{-1})$

SEZNAM OBRÁZKŮ

Obrázek 1. Postup při experimentální identifikaci	12
Obrázek 2. Příklady různých typů vstupních signálů	16
Obrázek 3. Chyba výstupu	18
Obrázek 4. Chyba vstupu	18
Obrázek 5. Chyba rovnice	19
Obrázek 6. Model stochastického objektu	25
Obrázek 7. Blokové schéma ARX modelu	27
Obrázek 8. Blokové schéma ARMAX modelu	32